Perbandingan Pre-Processing Opini Netizen Terhadap RUU PKS Menggunakan Algoritma Naive Bayes Classifier

Dina Maulina*[1], Mia Andhara Corry[2]
[1,2] Ilmu Komputer, Universitas AMIKOM Yogyakarta
E-mail: *[1]**dina.m@amikom.ac.id**, [2]mia.0771@students.amikom.ac.id

Abstrak

Proses pengesahan RUU-PKS menjadi topik pembicaraan dikalangan masyarakat khususnya sosial media, banyaknya opini yang disampaikan menyulitkan Netizen atau pengguna sosial media untuk membedakan opini negatif, positif ataupun netral. Dengan analisis sentimen opini acak tersebut dapat digabungkan dan diolah menjadi data sehingga menghasilkan sebuah informasi yang lebih jelas, pada proses sentimen analisis terdapat tahapan pre-processing. Untuk melakukan proses klasifikasi atau analisis data digunakan teknik machine learning bernama Naive Bayes sebuah metode yang dibuat untuk mengklasifikasikan data berbentuk text yang mampu dimanfaatkan dalam memprediksi suatu nilai dari sekumpulan data, namun penggunaan opini terhadap isu sebagai data penelitian menghasilkan jumlah data yang tidak seimbang atau imbalance sehingga diperlukan perhitungan akurasi dengan menggunakan metode confusion marix untuk meringkas kinerja klasifikasi yang memiliki data imbalance. Penelitian ini memiliki tujuan untuk mengetahui berapa nilai f1-score dari pre-processing opini Netizen terhadap RUU-PKS dengan menggunakan algoritma Naive Bayes Classifier, Mengetahui apakah tahapan pre-processing memiliki efektivitas dalam melakukan analisis sentimen pada opini Netizen terhadap RUU-PKS, Mengetahui berapa hasil perbandingan opini Netizen terhadap RUU-PKS dengan menggunakan kumpulan data cuitan pada media sosial Twitter serta mengetahui bagaimana hasil dari keseluruhan opini Netizen terhadap RUU-PKS pada sosial media Twitter. Penelitian ini menghasilkan empat kondisi pre-processing dengan 2.021 data opini dari Twitter mendapatkan nilai f1-score tertinggi pada kondisi preprocessing C yang tidak melakukan tahapan stopword namun melakukan tahapan normalisasi sebesar 72%.

Kata Kunci*— RUU-PKS, Analisis Sentimen, Pre-processing, Naive Bayes Classifier, Confusion Matrix*

1. PENDAHULUAN

RUU PKS merupakan salah satu topik yang kerap kali dibahas di Twitter ketika terjadi suatu kasus kekerasan seksual di Indonesia, tidak jarang banyaknya cuitan yang muncul dalam satu waktu membuat RUU PKS menjadi Trending Topic di Twitter Indonseia. Dengan sifat Twitter yang Microblogging menjadikan setiap cuitan yang disampaikan terlihat lebih ringkas, padat dan jelas, namun cuitan yang terus muncul dan tersusun secara acak menyebabkan kesulitan bagi Netizen untuk mengetahui opini negatif, positif ataupun netral [1], dengan menggunakan sentiment analysis informasi acak tersebut dapat digabungkan dan diolah menjadi data sehingga menghasilkan sebuah informasi yang lebih jelas. Pada proses sentiment analysis terdapat sebuah tahapan pre-processing, pada tahapan ini setiap data teks akan dilakukan pembersihan untuk mendapatkan data yang bersih dan jelas, terdapat beberapa tahapan dalam proses ini, diantaranya cleansing text, case folding, tokenisasi, remove stopword stemming, normalisasi, dan tahapan-tahapan lainnya. Masing-masing dari tahapan pre-processing memiliki peran dalam pengolahan data teks, dengan tujuan akhir yaitu mendapat nilai term yang jelas sehingga dapat mempermudah

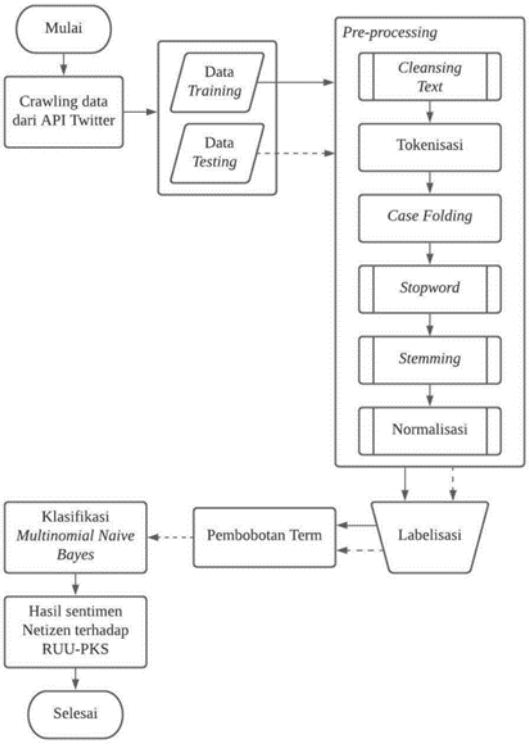
proses analisis sentimen saat melakukan klasifikasi ataupun meningkatkan besar akurasi model analisis. Dalam menganalisis sentimen terdapat sebuah teknik machine learning bernama Naive Bayes sebuah metode yang dibuat untuk mengklasifikasikan data berbentuk text [2], dan algoritma Naive Bayes sendiri dapat digunakan untuk memprediksi suatu nilai dari variabel [3]. Besar dari nilai akurasi pada proses klasifikasi dapat menentukan tingkat efektifitas sebuah proses ataupun algoritma namun jumlah data yang tidak seimbang atau imbalance tidak bisa menjadikan nilai akurasi sebagai nilai akurasi dari sebuah proses, confusion matrix dapat meringkas kinerja klasifikasi yang memiliki data imbalance menggunakan hasil nilai f-score sebagai pengganti nilai akurasi proses klasifikasi. Tujuan dari penelitian ini adalah Mengetahui berapa nilai f1-score dari pre-processing opini Netizen terhadap RUU-PKS dengan menggunakan algoritma Naive Bayes Classifier, Mengetahui apakah tahapan pre-processing memiliki efektivitas dalam melakukan analisis sentimen pada opini Netizen terhadap RUU-PKS, Mengetahui berapa hasil perbandingan opini Netizen terhadap RUU-PKS dengan menggunakan kumpulan data cuitan pada media sosial Twitter serta mengetahui bagaimana hasil dari keseluruhan opini Netizen terhadap RUU-PKS pada media sosial Twitter.

2. METODE PENELITIAN

Metode yang diajukan penulis untuk menentukan sentimen Netizen terhadap RUU-PKS terdiri dari beberapa proses. Rangkaian proses-proses yang akan dilakukan yaitu: pengumpulan data mengenai opini-opini Netizen terhadap RUU-PKS pada media sosial Twitter, kemudian kumpulan data opini akan di

pre-processing, hasil dari proses ini akan diberikan label penentuan sentimen atau proses labelisasi secara manaual, selanjutnya setiap kata pada hasil proses sebelumnya akan diberikan bobot menggunakan Term Frequency, setelah itu akan dilakukan proses klasifikasi menggunakan algoritma Naive Bayes sehingga didapatkan hasil analisis sentimen pada Netizen Twitter.

Pre-Processing merupakan proses menyeleksi data mentah yang akan diproses pada sebuah sistem untuk mendapatkan data yang berkualitas dan sudah terstruktur dengan jelas dengan mengurangi teks-teks secara signifikan atau teks yang tidak berpengaruh terhadap dokumen. Berikut ini adalah gambar arsitektur umum yang mendeskripsiskan setiap metodologi pada penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Arsitektur Umum

Manning menjelaskan bahwa probabilitas dari sebuah dokumen d yang ada pada kelas c dapat dihitung dengan persamaan 1 di bawah ini [5].

$$P(c|d) \propto P(c) \prod_{1 \le k \le n_d} P(t_k|c) \quad ...(1)$$

Keterangan:

P(c) : prior probability dari sebuah dokumen yang ada pada kelas c

$<t_1,t_2,\ldots,tn_d>$: kumpulan token yang digunakan untuk mengklasifikasi dan n_d merupakan jumlah token tersebut pada dokumen $_d$.

Untuk memperkirakan prior probability P(c) digunakan persamaan 2 sebagai berikut [5].

$$P(c) = \frac{Nc}{N} \quad ...(2)$$

Keterangan:

Nc : jumlah dari dokumen training pada kelas c

N : jumlah keseluruhan dokumen training dari seluruh kelas.

Untuk memperkirakan conditional probabilities P(t|t) digunakan persamaan 3, yaitu [5].

$$P(t_k|C) = \frac{Tct}{\Sigma\, t1 \epsilon v Tct1} \quad ....(3)$$

Keterangan:

T_{ct} : jumlah kemunculan term t dalam dokumen training dari kelas c

$\Sigma\, t^1 \in v\, T_{ct}{}^1$: jumlah total dari keseluruhan term yang terdapat dalam sebuah dokumen training dari kelas c.

Untuk menghilangkan nilai nol, digunakan add-one atau laplace smoothing. Proses ini menambahkan nilai satu (1) pada setiap nilai Tct dari perhitungan conditional probabilities. Sehingga untuk conditional probabilities menjadi seperti persamaan 4 di bawah [5].

$$P(t_k|c) = \frac{T_{c_t}+1}{(\sum t' \in v T_{c_{t'}})+B'} \quad \ldots (4)$$

Keterangan:

B' = jumlah keseluruhan term unik dari seluruh kelas

Untuk mendapatkan nilai akhir probabilitas pada dokumen data yang diuji, apakah dokumen uji tersebut termasuk dalam kelas negatif atau positif digunakan persamaan 5 [5].

$$P = \frac{N_c}{N} \times \frac{T_{c_t}+1}{(\sum t' \in v T_{c_{t'}})+B'} \quad \ldots (5)$$

Pengujian Akurasi pada penelitian ini mengguakan Confusion matrix dapat meringkas kinerja klasifikasi pengklasifikasi yang sehubungan dengan beberapa data uji, dengan menggunakan matrix dua dimensi, diindeks dalam satu dimensi oleh kelas sebenarnya dari suatu objek dan di dimensi lain oleh kelas yang ditetapkan dari pengklasifikasi [6]. Bentuk gambaran confusion matrix dapat dilihat pada tabel 1.

Tabel 1. Confusion Matrix

		Predicted	
Actual		Negative	Positive
	Negative	TN	TP
	Positive	FN	FP

Keterangan:

TN: True Negative TP: True Positive

FN: False Negative FP: False Negative

Keakuratan model (melalui confusion matrix) dihitung dengan menggunakan recall dan precision yang umum digunakan untuk mengklasifikasi. Precision menunjukkan seberapa akurat model untuk memprediksi nilai positif. Dengan demikian, mengukur keakuratan hasil positif yang diprediksi [7]. Recall digunakan untuk mengukur kekuatan suatu model dalam memprediksi hasil nilai positif [7], atau juga dikenal sebagai sensitivitas model. Rumus perhitungan precision dan recall dapat dilihat pada persamaan 6 dan 7 di bawah ini:

$$Recall = \frac{TP}{TP+FN} \quad \ldots (6)$$

$$Precision = \frac{TP}{TP+FP} \quad \ldots (7)$$

Terdapat metrik lain yang umum digunakan dalam pengaturan klasifikasi yaitu f1-score yang menggunakan hasil perhitungan precision dan recall dari pengklasifikasi. Untuk klasifikasi kasus positif, akan membantu untuk memahami tradeoff antara kebenaran dan cakupan [7]. Rumus umum untuk menghitung f1-score dapat dilihat pada persamaan 9 di bawah ini.

$$F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} \quad \ldots (8)$$

3. HASIL DAN PEMBAHASAN

3.1. Pembahasan

Sistem yang sudah dibangun dilakukan pemeriksaan atau pengujian sistem. Untuk menjelaskan proses pelatihan tehadap data training dan testing, diberikan 4 dokumen cuitan sebagai contoh perhitungan dengan menggunakan data testing sebagai data utamanya. Data yang digunakan sudah melalui tahap pre-processing. Dapat dilihat pada tabel 2 di bawah.

Tabel 2. Contoh Data

No	Sebelum Pre-Processing	Hasil Pre-Processing	Label
1	Kepada yang terhormat dan mulia; Pimpinan dan Anggota DPR RI Segera Sahkan RUU PKS !	[hormat, mulia, pimpin, anggota, dpr, sahkan, ruu, pks]	Positif
2	Udah 2021 loh, ruu pks mau dimasukin apa dikeluarin dari prolegnas lagi kaya taun kmrn?	[sudah, ruu, pks, masuk, keluar, prolegnas]	Netral
3	Nafsu banget buat menggagalkan RUU PKS	[nafsu, banget, gagal, ruu, pks]	Negatif
4	Selamat hari ibu utk semua perempuan yg ada di Indonesia. Semoga RUU PKS segera disahkan	[selamat, perempuan, Indonesia, moga, ruu, pks, sah]	?

Setelah melewati pre-processing kumpulan term pada dokumen dihitung dan dikumpulkan menjadi satu tabel. Kemunculan setiap term dapat dilihat pada tabel 3 di bawah.

Tabel 3. Daftar Term dan Frekuensi

No	Term	Term Frequency			
		D1	D2	D3	D4
1	hormat	1	0	0	0
2	mulia	1	0	0	0
3	pimpin	1	0	0	0
4	angota	1	0	0	0
5	dpr	1	0	0	0
6	sahkan	1	0	0	0
7	ruu	1	1	1	1
8	pks	1	1	1	1
9	sudah	0	1	0	0
10	masuk	0	1	0	0
11	Keluar	0	1	0	0
12	prolegnas	0	1	0	0
13	nafsu	0	0	1	0
14	banget	0	0	1	0
15	gagal	0	0	1	0
16	selamat	0	0	0	1
17	perempuan	0	0	0	1
18	indonesia	0	0	0	1
19	moga	0	0	0	1
20	sah	0	0	0	1

Setelah menghitung seluruh term yang muncul disetiap dokumen, setiap term akan dihitung dengan menggunakan Multinomial Naive Bayes untuk mengetahui klasifikasi dari dokumen D4.

Setelah data memalui tahap pre-processing dan menghitung jumlah term pada setiap dokumen (tabel 3), selanjutnya menghitung prior probabilitas dengan menggunakan persamaan 2:

1. P(positif) = 217/406
2. P(netral) = 97/406
3. P(negative) = 92/406

Keterangan:

Nilai 217 adalah jumlah sentimen positif pada data testing.
Nilai 97 adalah jumlah sentimen netral pada data testing.
Nilai 92 adalah jumlah sentimen negative pada data testing.
Nilai 406 adalah jumlah dokumen pada data testing.

Untuk mengetauhi seberapa banyak jumlah dokumen yang berlabel sama pada data testing menggunakan nilai prio, nilai ini nantinya akan dikalikan dengan nilai probabilitas kemunculan term. Hitungan ini menggunakan persamaan 3:

1. P(ruu|D1|positif) = 1/203
2. P(ruu|D2|netral) = 1/386
3. P(ruu|D3|negatif) = 1/358

Keterangan:

D1, D2, D3 adalah contoh dokumen yang ada pada data testing.
Nilai 1 adalah nilai term yang muncul pada 1 dokumen.
Nilai 203 adalah jumlah seluruh term yang ada pada label positif.

Untuk menghilangkan nilai 0 pada term digunakan perhitungan laplace smoothing menggunakan persamaan 4. Hasil dari proses ini akan menjadi model dari klasifikasi. Contoh perhitungan laplace smoothing term “gagal” pada dokumen 3 atau D3 pada:

1. Term (gagal|D3|positif) = (1+1)/(203+386) = 0.0034
2. Term (gagal|D3|netral) = (1+1)/(386+386) = 0. 00259
3. Term (gagal|D3|negatif) = (1+1)/(386+358) = 0. 00269

Hasil dari perhitungan nilai laplace smoothing pada seluruh term dapat dilihat pada tabel 4.

Tabel 4. Hasil Perhitungan Lapalce S Pada Term

No	Term	Term Frequency		
		Positif	Negatif	Netral
1	hormat	0.0034	0.00269	0.00259
2	mulia	0.0034	0.00269	0.00259
3	pimpin	0.0034	0.00269	0.00259
4	angota	0.0034	0.00269	0.00259
5	dpr	0.0034	0.00269	0.00259
6	sahkan	0.0034	0.00269	0.00259
7	ruu	0.0034	0.00269	0.00259
8	pks	0.0034	0.00269	0.00259
9	sudah	0.0034	0.00269	0.00259
10	masuk	0.0034	0.00269	0.00259
11	Keluar	0.0034	0.00269	0.00259
12	prolegnas	0.0034	0.00269	0.00259
13	nafsu	0.0034	0.00269	0.00259
14	banget	0.0034	0.00269	0.00259
15	gagal	0.0034	0.00269	0.00259
16	selamat	0.0034	0.0027	0.00259

17	perempuan	0.0034	0.0027	0.00259
18	indonesia	0.0034	0.0027	0.00259
19	moga	0.0034	0.0027	0.00259
20	sah	0.0034	0.0027	0.00259

Mulai dari sini data D4 akan mulai diklasifikasikan labelnya dengan menggunakan Multinomial Naive Bayes. Sebelum menghitung nilai laplace smoothing term pada D4 perlu dilakukannya proses match making, tabel 5 merupakan hasil matching menggunakan data utuh. Proses ini bertujuan untuk menemukan term yang sama-sama muncul antara term pada D4 dengan jumlah term yang ada pada data utuh atau data train. Apabila kata term muncul lebih dari satu kali maka nilai laplace smoothing akan dipangkatkan dengan jumlah term frequency berdasarkan kata yang sama dan kelas yang sama. Contoh, jika term 'selamat' memiliki tf sebanyak 2 kali. Pangkatkan nilai CP&LS-nya untuk menyederhanakan hitungan. Term (selamat|D4|positif) = 0.0034^2 = 0.002688172

Tabel 5. Nilai Match Making Pada D4

No	Term	Negatif	Positif	Netral
1	selamat	0	2	1
2	prerempuan	12	0	15
3	indoneia	7	24	5
4	moga	0	15	0
5	ruu	227	1269	231
6	pks	217	1187	212
7	sah	24	144	31

Setelah melakukan proses match making selanjutnya adalah mencari nilai dari laplace smoothing dengan term frequency. Hasil nilai dapat dilihat pada tabel 6 di bawah.

Tabel 6. Nilai Conditional P dan Lapalce S Pada D4

No	Term	Negatif	Positif	Netral
1	selamat	0.002688	1.153E-05	0.002591
2	prerempuan	1.42E-31	0.0033956	1.59E-39
3	indoneia	1.01E-18	5.52E-60	1.17E-13
4	moga	0.002688	9.199E-38	0.002591
5	ruu	0	0	0
6	pks	0	0	0
7	sah	2.03E-62	0	6.54E-81
Total		0.005376	0.003407	0.005181

Berikutnya menentukan nilai klasifikasi atau probabilitas dari D4 adalah dengan mengkalikan nilai total dari laplace smoothing dan term frequency dengan nilai prior probabilitas menggunakan persamaan 5.

1. Probabilitas D4 terhadap kelas negatif:
   P (negatif|D4) = 92/406*0.005376= 0.0012
2. Probabilitas D4 terhadap kelas positif:
   P (netral|D4) = 97/406*0.003407= 0.0018
3. Probabilitas D4 terhadap kelas netral:
   P (positif|D4) = 217/406*0.005181= 0.0012

Dari perhitungan di atas dapat dilihat bahwa nilai tertinggi adalah dari label Positif, maka nilai sentimen dari dokumen 4 atau D4 adalah Positif.

3.2. Hasil Evaluasi Sistem

Akurasi pengujian sistem dengan menggunakan f1-score, sistem melakukan analisis sebanyak 4 kali dengan menggunakan 4 kondisi berbeda yang sebelumnya sudah dijelaskan pada bagian 2.8. Dilakukannya 4 kali percobaan dengan harapan bisa mendapatkan nilai akurasi yang baik dari masing-masing kondisi. Hasil pengujian pada masing-masing kondisi dapat dilihat pada tabel 7.

Tabel 7. Hasil Pengujian ke-4 Kondisi

No			Kondisi Percobaan											
			Predicted											
			A			B			C			D		
			-1	0	1	-1	0	1	-1	0	1	-1	0	1
1	Actual	-1	39	0	18	37	1	28	46	1	23	32	1	35
		0	5	10	39	2	14	39	4	15	41	3	12	37
		1	6	1	287	3	0	281	1	2	272	4	2	279
2		-1	41	0	36	33	1	24	37	0	29	40	1	24
		0	4	16	48	5	17	50	5	8	54	5	11	40
		1	2	1	257	4	0	275	2	1	269	2	0	284
3		-1	40	0	34	33	0	31	39	0	30	33	0	31
		0	3	8	44	6	12	41	3	15	35	4	14	42
		1	1	1	274	2	0	280	3	1	279	0	1	280
4		-1	40	0	19	35	1	27	41	0	18	32	0	29
		0	9	11	44	3	15	39	3	17	41	5	17	46
		1	6	1	275	3	0	282	1	0	284	2	0	274

Pada tabel 7 menampilkan semua hasil dari kalkulasi program dari 4 kali percobaan dengan berbagai kondisi yang sudah dijelaskan pada 2.9 sebelumnya. Selanjutnya untuk mengetahui nilai f1-score secara keseluruhan, perlu dilakukannya penacarian nilai precision dan recall. Untuk mencari nilai tersebut digunakan persamaan 6 dan 7. Dikarenakan pada penelitian ini menggunakan 3 kelas maka nilai FP dihitung menurun dari tabel sedangkan nilai FN dihitung mendatar dari tabel. Contoh perhitungan recall kondisi A pada Percobaan 1:

1. Kelas Negative = $Precision = \frac{TP}{TP+FP} = \frac{39}{39+(5+6)} = 0.78$
2. Kelas Neutral = $\frac{10}{10+(0+1)} = 0.90$
3. Kelas Positive = $\frac{287}{287+(18+39)} = 0.83$

Dikarenakan tedapat 3 kelas maka hasil dari seluruh recall setiap kelas dijumlahkan lalu dibagi dengan jumlah kelasnya.

Precision Kondisi A $= \frac{\boldsymbol{precision\ Negative + precision\ Neutral + precision\ Positive}}{\boldsymbol{Jumlah\ kelas}}$

$= \frac{\mathbf{0.78+0.90+0.83}}{\mathbf{3}}$

$= \mathbf{0.83}$

Contoh perhitungan precision kondisi A pada percobaan 1:

1. Kelas Negative = $Recall = \frac{TP}{TP+FN} = \frac{39}{39+(0+18)} = 0.68$
2. Kelas Neutral = $\frac{10}{10+(5+39)} = 0.18$
3. Kelas Positive = $\frac{287}{287+(6+1)} = 0.97$

Dikarenakan tedapat 3 kelas maka hasil dari seluruh recall setiap kelas dijumlahkan lalu dibagi dengan jumlah kelasnya.

Recall Kondisi A $= \frac{\boldsymbol{recall\ Negative + recall\ Neutral + recall\ Positive}}{\boldsymbol{Jumlah\ kelas}}$

$= \frac{\mathbf{0.68+0.18+0.97}}{\mathbf{3}}$

$= \mathbf{0.61}$

Hasil perhitungan dari precision dan recall seluruh data pada tabel 7 dapat dilihat pada tabel 8.

Tabel 8. Nilai Precision dan Recall

Nomor Percobaan		Kondisi Percobaan			
		A	B	C	D
1	*Precision*	0.83	0.87	0.86	0.80
	Recall	0.61	0.60	0.63	0.56
2	*Precision*	0.85	0.60	0.83	0.86
	Recall	0.59	0.84	0.56	0.60
3	*Precision*	0.86	0.57	0.87	0.87
	Recall	0.56	0.87	0.61	0.58
4	*Precision*	0.82	0.87	0.91	0.87
	Recall	0.61	0.60	0.66	0.59

Setelah mendapatkan nilai dari recall dan precision, dilakukan perhitungan dengan menggunakan persamaan 8 untuk mendapatkan nilai f1-score.

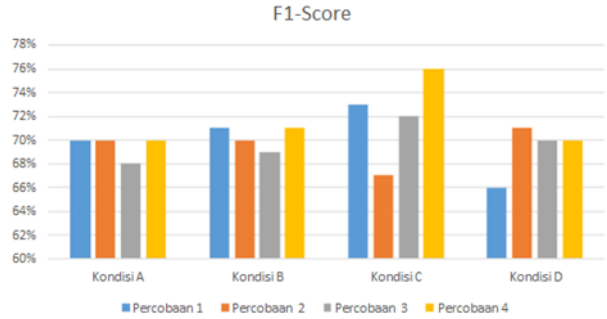
Contoh perhitungan f1-score pada Kondisi A:

Percobaan 1 = $F_1 = 2 \times \frac{Precision \times Recall}{Precision + Recall} = 2 \times \frac{0{,}83 \times 0{,}61}{0{,}83 + 0{,}61} = 0{,}70$

Selanjutnya seluruh nilai dari berbagai kondisi dijumlahkan untuk menedapatkan nilai rata-rata f1-score dari empat kali percobaan klasifikasi program. Untuk nilai rata-rata f1-score dari masing-masing kondisi dapat dilihat pada tabel 9 dan gambar 1.

Tabel 9. Nilai Rata-Rata F-Score

Kondisi Percobaan	Nomor Percobaan				Rata-rata F1-Score
	1	2	3	4	
A	0.70	0.70	0.68	0.70	0.69
B	0.71	0.70	0.69	0.71	0.70
C	0.73	0.67	0.72	0.76	0.72
D	0.66	0.71	0.70	0.70	0.69



Gambar 2. Hasil Perbandiangan F1-Score

4. KESIMPULAN

Dari tabel 9, gambar 1 dan berdasarkan kondisi yang sudah dijelaskan pada sub bab 2.8 dapat dilihat bahwa pada model A dengan kondisi tanpa menggunakan stopword dan normalisasi berlangsung stabil namun mengalami penurunan pada percobaan ke-3 yaitu 68%. Pada model B dengan kondisi melalui proses stopword namun tidak mendapati proses normalisasi juga mengalami penuruan hingga 69% namun pada percobaan ke-4 kembali sama seperti percobaan pertama yaitu 71%. Selanjutnya pada model C dengan kondisi tidak melalui proses stopword namun melewati proses normalisasi mendapati kualitas nilai f1-score pada percobaan ke-2 mengalami penurunan yang drastis namun pada percobaan berikutnya terus mengalami peningkatan nilai f1-score 76%. Terakhir pada model D dengan kondisi data melalui proses stopword dan juga normalisasi mendapati nilai terendah f1-score sebesar 66% dan kemudian pada ke-2 mendapati peningkatan namun seterusnya tingkat presentase stabil sampai dengan percobaan ke-4. Secara nilai keseluruhan model C dengan kondisi data tidak melalui proses stopword namun melalui proses normalisasi memiliki nilai rata-rata f1-score tertinggi, yaitu sebesar 76%.

Kesimpulan, proses stopword dan normalisasi atau proses pre-processing tidak selalu menjadi poin utama dalam meningkatkan performa program analisis sentimen namun kualitas data dan term yang jelas dan pelabelan data yang tepat tentu dapat menjadi kunci untama dalam kesuksesan analisis sentimen.

DAFTAR PUSTAKA

[1] A. K. Wardadi, G. P. Manurung and N. F. Rais, "Analisis Keberlakuan RKUHP dan RUU-PKS dalam Mengatur Tindak Kekerasan Seksual," Lex Scientia Law Review, pp. 30-39, 2019.

[2] A. Saputra, "SURVEI PENGGUNAAN MEDIA SOSIAL DI KALANGAN MAHASISWA KOTA PADANG MENGGUNAKAN TEORI USES AND GRATIFICATIONS," BACA: Jurnal Dokumentasi dan Informasi, pp. 207- 216, 2019.

[3] S. Fransiska and Yolanda, "ANALISIS SENTIMEN TWITTER UNTUK REVIEW FILM MENGGUNAKAN ALGORITMA NAIVE BAYES CLASSIFIER (NBC) PADA SENTIMEN R PROGRAMMING," Jurnal Siliwangi, pp. 68-71, 2019.

[4] A. S. Widagdo, B. S. W. A. and A. Nasiri, "ANALISIS TINGKAT KEPOPULERAN E-COMERCE DI INDONESIA BERDASARKAN SENTIMENT SOSIAL MEDIA MENGGUNAKAN METODE NAIVE BAYES," Jurnal INFORMA Politeknik Indonusa Surakarta, pp. 1-5, 2020.

[5] M. A. Nurrohmat and Y. S. Nugroho, "Aplikasi Pemrediksi Masa Studi dan Predikat Kelulusan Mahasiswa Informatika Universitas Muhammadiyah Surakarta Menggunakan Metode Naive Bayes," Jurnal Ilmu Komputer dan Informatika, pp. 29-34, 2015.
[6] L. Dey, S. Chakraborty, A. Biswas, B. Bose and S. Tiwari, "Sentiment Analysis of Review Datasets using Naive Bayes and K-NN Classifier," arXiv Informatian Retrieval; Computation and Language, vol. 8, no. 4, pp. 54-62, 2016.
[7] I. F. Rozi, E. N. Hamdana and M. B. I. Alfahmi, "Pengembangan Aplikasi Analisis Sentimen Twitter Menggunakan Metode Naive Bayes Classifier," Jurnal Informatika Polinema, pp. 149-154, 2018.
[8] M. Syarifuddin, "Analisis Sentimen Opini Publik Terhadap Efek PSBB Pada 109Twitter Dengan Algoritma Decision Tree-KNN-Naive Bayes," Inti Nusa Mandiri, vol. 15, pp. 87-94, 2020.
[9] I. Zulfa and E. Winarko, "Sentimen Analisis Tweet Berbahasa Indonesia dengan Deep Belief Network," IJCCS, vol. 11, pp. 187-198, 2017.
[10] K. Sharma and A. Sambyal, "SENTIMENT ANALYSIS USING AMAZON DATA FOR WDE-KNN ALGORITHM," INTERNATIONAL JOURNAL OF INFORMATION AND COMPUTING SCIENCE, vol. 6, no. 2, pp. 1-9, 2019.
[11] L. Yang, Y. Li, J. Wang and R. S. Sherratt, "Sentiment Analysis for ECommerce Product Reviews in Chinese Based on Sentiment Lexicon and Deep Learning," Institute of Electrical and Electronics Engineers Access, vol. 8, pp. 23522-23530, 2020.
[12] "Rancangan Undang-Undang," Pusat Perancangan Undang-Undang Badan Keahlian DPR RI, [Online]. Available: https://pusatpuu.dpr.go.id/produk/index-draft-ruu. [Accessed 20 June 2021].
[13] H. Jogiyanto, Analisis dan Desain Sistem Informasi: Pendekatan Terstruktur Teori dan Praktek Aplikasi Bisnis, Yogyakarta: Andi Offest, 1995.
[14] H. A. Fatta, Analisis & Perancangan Sistem Informasi untuk Keunggulan Bersaing & Organisasi Modern, Yogyakarta: Penerbit ANDI, 2007.
[15] T. Nasukawa and Y. Jeonghee, "Sentiment Analysis: Capturing favorability using Natural Language Processing," Proceedings of the 2nd International Conference on Knowledge Capture, pp. 70-77, 2003.
[16] B. Liu, "Sentient Analysis and Opinion Mining," Morgan & Claypool Publishers, May 2012.
[17] Balya, "Analisis Sentimen Pengguna YouTube di Indonesia Pada Review Smartphone Menggunakan Naive Bayes," in Skripsi, Fakultas Ilmu Komputer dan Teknologi, Ilmu Komputer, Medan, Universitas Sumatera Utara, 2019.
[18] R. A. Simanjuntak, "Analisis Sentimen Pada Layanan Gojek Indonesia Menggunakan Multinomial Naive Bayes," in Skripsi, Fakultas Ilmu Komputer dan Teknologi, Ilmu Komputer, Medan, Universitas Sumatera 110Utara, 2018.
[19] R. Nasrullah, Media Sosial Perspektif Komunikasi, Budaya, dan Sosioteknologi, Bandung: Simbiosa Rekatama Media, 2015.
[20] E. Prasetyo, Data Mining - Mengolah Data Menjadi Informasi Menggunakan Matlab, Yogyakarta: ANDI Offset, 2014.
[21] Kusrini and E. T. Luthfi, Algoritma Data Mining, Yogyakarta: ANDI, 2009.
[22] R. T. Vulandari, DATA MINING Teori dan Aplikasi Rapidminer, Yogyakarta: Gava Media, 2017.
[23] R. Feldman and J. Sanger, The Text Mining Handbook : Advanced Approaches in Analyzing Unstructured Data, New York: Cambridge University Press, 2007.

[24] S. Institute, Getting Started with SAS® Text Miner 4.2, North Calorina: SAS Publishing, 2010.
[25] W. Oktinas, "Analisis Sentimen Pada Acara Televisi Menggunakan Improverd K-Nearest Neighbor," in Skripsi, Fakultas Ilmu Komputer dan Teknologi, Ilmu Komputer, Medan, Universitas Sumatra Utara, 2017.
[26] D. Maulina and R. Sagara, "KLASIFIKASI ARTIKEL HOAX MENGGUNAKAN SUPPORT VECTOR MACHINE LINEAR DENGAN PEMBOBOTAN TERM FREQUENCY – INVERSE DOCUMENT FREQUENCY," Jurnal Mantik Penusa, vol. II, pp. 35-40, 2018.
[27] "What is Python," Python, [Online]. Available: https://docs.python.org/3/faq/general.html#what-is-python. [Accessed 20 Juni 2021].
[28] Jupyter, [Online]. Available: https://jupyter.org/about. [Accessed 20 Juni 2021].
[29] C. D. Manning, P. Raghavan and H. Schütze, An Introduction to Information Retrieval, Cambridge, England: Cambridge University Press, 2009.
[30] T. K.M., "Confusion Matrix. In: Sammut C.," Webb G.I. (eds) Encyclopedia of Machine Learning., [Online]. Available: https://link.springer.com/referenceworkentry/10.1007%2F978-0-387-30164-1118_157#howtocite. [Accessed 24 August 2021].
[31] A. Kulkarni, D. Chong and F. A. Batarseh, "Foundations of Data Imbalance and Solution for a Data Democracy," Data Democracy At the Nexus of Artificial Intelligence, Software Development, and Knowledge Engineering, pp. 83-106, 2020.